We claim:

1. A computer program product embodied on computer readable media readable by a computing system in a computing environment, for enforcing security policy using style sheet processing, comprising:

an input document;

one or more stored policy enforcement objects, wherein each of said stored policy enforcement objects specifies a security policy to be associated with zero or more elements of said input document;

a Document Type Definition (DTD) corresponding to said input document, wherein said DTD has been augmented with one or more references to selected ones of said stored policy enforcement objects;

an augmented style sheet processor, wherein said augmented processor further comprises:

computer-readable program code means for loading said DTD;

computer-readable program code means for resolving each of said one or more references in said loaded DTD;

computer-readable program code means for instantiating said policy enforcement objects associated with said resolved references;

computer-readable program code means for executing selected ones of said instantiated policy enforcement objects during application of one or more style sheets to said input document, wherein a result of said computer-readable program code means for executing is an interim transient document reflecting said execution;

computer-readable program code means for generating one or more random encryption keys;

computer-readable program code means for encrypting selected elements of said interim transient document, wherein a particular one of said generated random encryption keys may be used to encrypt one or more of said selected elements, while leaving zero or more other elements of said interim transient document unencrypted;

computer-readable program code means for encrypting each of said one or more random encryption keys; and

computer-readable program code means for creating an encrypted output document comprising said zero or more other unencrypted elements, said selected encrypted elements, and said encrypted encryption keys;

computer-readable program code means for requesting said encrypted output document by a key recovery agent;

computer-readable program code means for receiving said requested output document; and

an augmented document processor, comprising:

computer-readable program code means for decrypting each of said encrypted encryption keys; and

computer-readable program code means for decrypting said requested output document using said decrypted keys, thereby creating a result document.

2. The computer program product according to Claim 1, further comprising computer-readable program code means for rendering said result document on said client device.

3. The computer program product according to Claim 1, wherein said interim transient document comprises one or more encryption tags identifying elements needing encryption.

4. The computer program product according to Claim 1, wherein said input document is specified in an Extensible Markup Language (XML) notation.

5. The computer program product according to Claim 4, wherein said result document is specified in said XML notation.

6. The computer program product according to Claim 1, wherein said stored policy enforcement objects further comprise computer-readable program code means for overriding a method for evaluating said elements of said input document, and wherein said computer-readable program code means for executing further comprises computer-readable program code means for executing said computer-readable program code means for overriding.

7. The computer program product according to Claim 6, wherein said style sheets are specified in an Extensible Stylesheet Language (XSL) notation.

8. The computer program product according to Claim 7, wherein said method is a value-of method of said XSL notation, and wherein said computer-readable program code means for overriding said value-of method is by subclassing said value-of method.

9. The computer program product according to Claim 6 or Claim 8, wherein:

said overridden method comprises:

computer-readable program code means for generating encryption tags; and

computer-readable program code means for inserting said generated encryption tags into said interim transient document to surround elements of said interim transient document which are determined to require encryption; and

said computer-readable program code means for encrypting selected elements encrypts those elements surrounded by said inserted encryption tags.

10. The computer program product according to Claim 2, wherein:

each of said instantiated policy enforcement objects further comprises:

a specification of a community that is authorized to view said elements associated with said security policy, said specification of said communities further comprising specification of at least one of: (1) one or more individual users or processes which are community members, and (2) one or more groups which are community members, wherein each of said groups comprises one or more individual users or processes; and

an encryption requirement for said elements associated with said security policy.

11. The computer program product according to Claim 10, wherein said encryption requirement further comprises specification of an encryption algorithm.

12. The computer program product according to Claim 10, wherein said encryption requirement further comprises specification of an encryption algorithm strength value.

13. The computer program product according to Claim 10, wherein:

said computer-readable program code means for encrypting said encryption keys further comprises:

computer-readable program code means for encrypting a different version of each of said random encryption keys for each of said one or more members of each of zero or more of said communities which uses said encryption key, and wherein each of said different versions is encrypted using a public key of said community member for which said different version was encrypted; and

computer-readable program code means for ensuring that said key recovery agent is one of said members of each of said communities, thereby ensuring that one of said different versions is encrypted using said public key of said key recovery agent.

14. The computer program product according to Claim 10, wherein said encryption requirement may have a null value to indicate that said specified security policy does not require encryption.

15. The computer program product according to Claim 1, wherein said computer-readable program code means for encrypting selected elements uses a cipher block chaining mode encryption process.

16. The computer program product according to Claim 13, further comprising:

computer-readable program code means for creating a key class for each unique community, wherein said key class is associated with each of said encrypted elements for which this unique community is an authorized viewer, and wherein said key class comprises: (1) a strongest encryption requirement of said associated encrypted elements; (2) an identifier of each of said members of said unique community; and (3) one of said different versions of said encrypted encryption key for each of said identified community members; and

wherein:

said computer-readable program code means for generating said one or more random encryption keys generates a particular one of said random encryption keys for each of said key classes, and wherein each of said different versions in a particular key class is encrypted from said generated encryption key generated for said key class; and

said computer-readable program code means for encrypting selected elements uses that one of said particular random encryption keys which was generated for said key class with which said selected element is associated.

17. The computer program product according to Claim 13, wherein:

said computer-readable program code means for decrypting said requested output document further comprises:

computer-readable program code means for decrypting, for each of said communities, said different version of said random encryption key which was encrypted using said public key of said key recovery agent, wherein said computer-readable program code means for decrypting uses a private key of said key recovery agent, thereby creating a decrypted key for each of said communities; and

computer-readable program code means for decrypting each of said encrypted elements in said requested output document using said decrypted keys; and

said computer-readable program code means for rendering further comprises:

computer-readable program code means for rendering said decrypted elements and said other unencrypted elements.

18. The computer program product according to Claim 16, wherein:

said computer-readable program code means for decrypting said requested output document further comprises:

computer-readable program code means for decrypting, for each of said key classes, said different version of said random encryption key in said key class which was encrypted using said public key of said key recovery agent, wherein said computer-readable program code means for decrypting uses a private key of said key recovery agent which is associated with said public key which was used for encryption, thereby creating a decrypted key; and

computer-readable program code means for decrypting each of said encrypted elements in said requested output document using said decrypted keys; and

said computer-readable program code means for rendering further comprises:

computer-readable program code means for rendering said decrypted elements and said other unencrypted elements.

19. The computer program product according to Claim 1, wherein said DTD is replaced by a schema.

20. The computer program product according to Claim 10, wherein said encryption requirement further comprises specification of an encryption key length.

21. The computer program product according to Claim 9, wherein said inserted encryption tags may surround either values of said elements or values and tags of said elements.

22. A system for enforcing security policy using style sheet processing in a computing environment, comprising:

an input document;

one or more stored policy enforcement objects, wherein each of said stored policy enforcement objects specifies a security policy to be associated with zero or more elements of said input document;

a Document Type Definition (DTD) corresponding to said input document, wherein said DTD has been augmented with one or more references to selected ones of said stored policy enforcement objects;

an augmented style sheet processor, wherein said augmented processor further comprises:

means for loading said DTD;

means for resolving each of said one or more references in said loaded DTD;

means for instantiating said policy enforcement objects associated with said resolved references;

means for executing selected ones of said instantiated policy enforcement objects during application of one or more style sheets to said input document, wherein a result of said means for executing is an interim transient document reflecting said execution;

means for generating one or more random encryption keys;

means for encrypting selected elements of said interim transient document, wherein a particular one of said generated random encryption keys may be used to encrypt one or more of said selected elements, while leaving zero or more other elements of said interim transient document unencrypted;

means for encrypting each of said one or more random encryption keys; and

means for creating an encrypted output document comprising said zero or more other unencrypted elements, said selected encrypted elements, and said encrypted encryption keys;

means for requesting said encrypted output document by a key recovery agent;

means for receiving said requested output document; and

an augmented document processor, comprising:

means for decrypting each of said encrypted encryption keys; and

means for decrypting said requested output document using said decrypted keys, thereby creating a result document.

23. The system according to Claim 22, further comprising means for rendering said result document on said client device.

24. The system according to Claim 22, wherein said interim transient document comprises one or more encryption tags identifying elements needing encryption.

25. The system according to Claim 22, wherein said input document is specified in an Extensible Markup Language (XML) notation.

26. The system according to Claim 25, wherein said result document is specified in said XML notation.

27. The system according to Claim 22, wherein said stored policy enforcement objects further comprise means for overriding a method for evaluating said elements of said input document, and wherein said means for executing further comprises means for executing said computer-readable program code means for overriding.

28. The system according to Claim 27, wherein said style sheets are specified in an Extensible Stylesheet Language (XSL) notation.

29. The system according to Claim 28, wherein said method is a value-of method of said XSL notation, and wherein said means for overriding said value-of method is by subclassing said value-of method.

30. The system according to Claim 27 or Claim 29, wherein:

said overridden method comprises:

means for generating encryption tags; and

means for inserting said generated encryption tags into said interim transient document to surround elements of said interim transient document which are determined to require encryption; and

said means for encrypting selected elements encrypts those elements surrounded by said inserted encryption tags.

31. The system according to Claim 23, wherein:

each of said instantiated policy enforcement objects further comprises:

a specification of a community that is authorized to view said elements associated with said security policy, said specification of said communities further comprising specification of at least one of: (1) one or more individual users or processes which are community members, and

(2) one or more groups which are community members, wherein each of said groups comprises one or more individual users or processes; and

an encryption requirement for said elements associated with said security policy.

32. The system according to Claim 31, wherein said encryption requirement further comprises specification of an encryption algorithm.

33. The system according to Claim 31, wherein said encryption requirement further comprises specification of an encryption algorithm strength value.

34. The system according to Claim 31, wherein:

said means for encrypting said encryption keys further comprises:

means for encrypting a different version of each of said random encryption keys for each of said one or more members of each of zero or more of said communities which uses said encryption key, and wherein each of said different versions is encrypted using a public key of said community member for which said different version was encrypted; and

means for ensuring that said key recovery agent is one of said members of each of said communities, thereby ensuring that one of said different versions is encrypted using said public key of said key recovery agent.

35. The system according to Claim 31, wherein said encryption requirement may have a null value to indicate that said specified security policy does not require encryption.

36. The system according to Claim 22, wherein said means for encrypting selected elements uses a cipher block chaining mode encryption process.

37. The system according to Claim 34, further comprising:

means for creating a key class for each unique community, wherein said key class is associated with each of said encrypted elements for which this unique community is an authorized viewer, and wherein said key class comprises: (1) a strongest encryption requirement of said associated encrypted elements; (2) an identifier of each of said members of said unique community; and (3) one of said different versions of said encrypted encryption key for each of said identified community members; and

wherein:

said means for generating said one or more random encryption keys generates a particular one of said random encryption keys for each of said key classes, and wherein each of said different versions in a particular key class is encrypted from said generated encryption key generated for said key class; and

said means for encrypting selected elements uses that one of said particular random encryption keys which was generated for said key class with which said selected element is associated.

38. The system according to Claim 34, wherein:

said means for decrypting said requested output document further comprises:

means for decrypting, for each of said communities, said different version of said random encryption key which was encrypted using said public key of said key recovery agent, wherein said means for decrypting uses a private key of said key recovery agent, thereby creating a decrypted key for each of said communities; and

means for decrypting each of said encrypted elements in said requested output document using said decrypted keys; and

said means for rendering further comprises:

means for rendering said decrypted elements and said other unencrypted elements.

39. The system according to Claim 37, wherein:

said means for decrypting said requested output document further comprises:

means for decrypting, for each of said key classes, said different version of said random encryption key in said key class which was encrypted using said public key of said key recovery agent, wherein said means for decrypting uses a private key of said key recovery agent which is associated with said public key which was used for encryption, thereby creating a decrypted key; and

means for decrypting each of said encrypted elements in said requested output document using said decrypted keys; and

said means for rendering further comprises:

means for rendering said decrypted elements and said other unencrypted elements.

40. The system according to Claim 22, wherein said DTD is replaced by a schema.

41. The system according to Claim 31, wherein said encryption requirement further comprises specification of an encryption key length.

42. The system according to Claim 30, wherein said inserted encryption tags may surround either values of said elements or values and tags of said elements.

43. A method for enforcing security policy using style sheet processing in a computing environment, comprising the steps of:

providing an input document;

providing one or more stored policy enforcement objects, wherein each of said stored policy enforcement objects specifies a security policy to be associated with zero or more elements of said input document;

providing a Document Type Definition (DTD) corresponding to said input document, wherein said DTD has been augmented with one or more references to selected ones of said stored policy enforcement objects;

executing an augmented style sheet processor, further comprising the steps of:

loading said DTD;

resolving each of said one or more references in said loaded DTD;

instantiating said policy enforcement objects associated with said resolved references;

executing selected ones of said instantiated policy enforcement objects during application of one or more style sheets to said input document, wherein a result of said step of executing is an interim transient document reflecting said execution;

generating one or more random encryption keys;

encrypting selected elements of said interim transient document, wherein a particular one of said generated random encryption keys may be used to encrypt one or more of said selected elements, while leaving zero or more other elements of said interim transient document unencrypted;

encrypting each of said one or more random encryption keys; and

creating an encrypted output document comprising said zero or more other unencrypted elements, said selected encrypted elements, and said encrypted encryption keys;

requesting said encrypted output document by a key recovery agent;

receiving said requested output document; and

executing an augmented document processor, further comprising the steps of:

decrypting each of said encrypted encryption keys; and

decrypting said requested output document using said decrypted keys, thereby creating a result document.

44. The method according to Claim 43, further comprising the step of rendering said result document on said client device.

45. The method according to Claim 43, wherein said interim transient document comprises one or more encryption tags identifying elements needing encryption.

46. The method according to Claim 43, wherein said input document is specified in an Extensible Markup Language (XML) notation.

47. The method according to Claim 46, wherein said result document is specified in said XML notation.

48. The method according to Claim 43, wherein said stored policy enforcement objects further comprise executable code for overriding a method for evaluating said elements of said input document, and wherein said executing selected ones step further comprises overriding said method for evaluating.

49. The method according to Claim 48, wherein said style sheets are specified in an Extensible Stylesheet Language (XSL) notation.

50. The method according to Claim 49, wherein said method is a value-of method of said XSL notation, and wherein said step of overriding said value-of method is by subclassing said value-of method.

51. The method according to Claim 48 or Claim 50, wherein:

said step of overriding further comprises the steps of:

generating encryption tags; and

inserting said generated encryption tags into said interim transient document to surround elements of said interim transient document which are determined to require encryption; and

said step of encrypting selected elements encrypts those elements surrounded by said inserted encryption tags.

52. The method according to Claim 44, wherein:

each of said instantiated policy enforcement objects further comprises:

a specification of a community that is authorized to view said elements associated with said security policy, said specification of said communities further comprising specification of at least one of: (1) one or more individual users or processes which are community members, and (2) one or more groups which are community members, wherein each of said groups comprises one or more individual users or processes; and

an encryption requirement for said elements associated with said security policy.

53. The method according to Claim 52, wherein said encryption requirement further comprises specification of an encryption algorithm.

54. The method according to Claim 52, wherein said encryption requirement further comprises specification of an encryption algorithm strength value.

55. The method according to Claim 52, wherein:

said step of encrypting said encryption keys further comprises the steps of:

encrypting a different version of each of said random encryption keys for each of said one or more members of each of zero or more of said communities which uses said encryption key, and wherein each of said different versions is encrypted using a public key of said community member for which said different version was encrypted; and

ensuring that said key recovery agent is one of said members of each of said communities, thereby ensuring that one of said different versions is encrypted using said public key of said key recovery agent.

56. The method according to Claim 52, wherein said encryption requirement may have a null value to indicate that said specified security policy does not require encryption.

57. The method according to Claim 43, wherein said step of encrypting selected elements uses a cipher block chaining mode encryption process.

58. The method according to Claim 55, further comprising the step of:

creating a key class for each unique community, wherein said key class is associated with each of said encrypted elements for which this unique community is an authorized viewer, and wherein said key class comprises: (1) a strongest encryption requirement of said associated encrypted elements; (2) an identifier of each of said members of said unique community; and (3)

one of said different versions of said encrypted encryption key for each of said identified community members; and

wherein:

said step of generating said one or more random encryption keys generates a particular one of said random encryption keys for each of said key classes, and wherein each of said different versions in a particular key class is encrypted from said generated encryption key generated for said key class; and

said step of encrypting selected elements uses that one of said particular random encryption keys which was generated for said key class with which said selected element is associated.

59. The method according to Claim 55, wherein:

said step of decrypting said requested output document further comprises the steps of:

decrypting, for each of said communities, said different version of said random encryption key which was encrypted using said public key of said key recovery agent, wherein said step of decrypting uses a private key of said key recovery agent, thereby creating a decrypted key for each of said communities; and

decrypting each of said encrypted elements in said requested output document using said decrypted keys; and

said step of rendering further comprises the step of:

rendering said decrypted elements and said other unencrypted elements.

60. The method according to Claim 58, wherein:

said step of decrypting said requested output document further comprises the steps of:

decrypting, for each of said key classes, said different version of said random encryption key in said key class which was encrypted using said public key of said key recovery agent, wherein said step of decrypting uses a private key of said key recovery agent which is associated with said public key which was used for encryption, thereby creating a decrypted key; and

decrypting each of said encrypted elements in said requested output document using said decrypted keys; and

said step of rendering further comprises the step of:

rendering said decrypted elements and said other unencrypted elements.

61. The method according to Claim 43, wherein said DTD is replaced by a schema.

62. The method according to Claim 52, wherein said encryption requirement further comprises specification of an encryption key length.

63. The method according to Claim 51, wherein said inserted encryption tags may surround either values of said elements or values and tags of said elements.